i have recently covered statistics by statquest. give me some 10 questions to do in python with dataset links to make great understanding of statistics. unformly distributre the levels og easy/medium and hard

Here are 10 progressively mixed easy/medium/hard problems, each with a concrete dataset link and a short Python-oriented task description. Levels are indicated as E/M/H.

1. One-variable distributions (E)

- **Dataset:** Titanic passengers (CSV) – Kaggle: https://www.kaggle.com/c/titanic/data[1]
- **Task:**
  - Load `train.csv` with pandas.
  - For `Age` and `Fare`, compute mean, median, mode, variance, standard deviation, and interquartile range.
  - Plot histogram and boxplot for each and comment on skewness using sample skewness formula and visual intuition.

2. Categorical proportions & barplots (E)

- **Dataset:** Penguins size data – GitHub (`penguins_size.csv`): https://github.com/allisonhorst/palmerpenguins/tree/master/inst/extdata[2]
- **Task:**
  - Compute frequency and relative frequency tables for `species` and `island`.
  - Compute conditional proportion of species within each island (e.g., P(species | island)).
  - Plot bar charts for marginal and conditional distributions; comment on any obvious dependence between species and island.

3. Correlation and scatterplots (E)

- **Dataset:** Advertising dataset (TV, Radio, Newspaper vs Sales) – GitHub: https://github.com/justmarkham/DAT8/blob/master/data/advertising.csv[2]
- **Task:**
  - Compute Pearson correlation matrix for all numeric variables.
  - Plot scatterplots of each ad channel vs `sales`.
  - Explain why high correlation does not prove causation and inspect at least one case where correlation is weaker.

4. Sampling distributions & CLT (M)

- **Dataset:** New York City Airbnb Open Data – Kaggle: https://www.kaggle.com/datasets/dgomonov/new-york-city-airbnb-open-data[1]
- **Task:**
  - Focus on one numeric column (e.g., `price`, after removing outliers and zeros).
  - Simulate sampling distributions of the sample mean for different sample sizes $n = 5, 30, 100$ by repeatedly drawing samples and storing the mean (e.g., 5,000 repetitions).
  - Plot histograms of these sampling distributions and show how the distribution becomes more normal and the variance shrinks as n increases.

5. Confidence intervals for a mean (M)

- **Dataset:** Students Performance in Exams – Kaggle: https://www.kaggle.com/datasets/spscientist/students-performance-in-exams[1]
- **Task:**
  - Use one score column, e.g. `math score`.
  - Compute a 95% confidence interval for the mean math score using:
    - Known-population-variance style with sample standard deviation as plug-in (normal approximation).
    - t-based confidence interval via `scipy.stats.t`.
  - Interpret the interval in plain language and compare the endpoints of the normal vs t-based intervals.

6. Hypothesis test for proportions (M)

- **Dataset:** Heart Disease UCI – Kaggle: https://www.kaggle.com/datasets/ronitf/heart-disease-uci[1]
- **Task:**
  - Turn `target` into "has disease" (1) vs "no disease" (0).
  - Test the hypothesis that the proportion of patients with heart disease is 0.5 using a one-sample z-test for a proportion.
  - Compute test statistic, p-value manually in Python (no black-box function), and conclude in context at $\alpha = 0.05$.

7. Simple linear regression (H)

- **Dataset:** Boston Housing (scikit-learn version – available via `sklearn.datasets.load_boston` is deprecated; use the open version instead):
  CSV mirror: https://github.com/selva86/datasets/raw/master/BostonHousing.csv[2]
- **Task:**
  - Fit a simple linear regression of `medv` (median house value) on `lstat` (lower status population).
  - Compute slope, intercept, residuals, R^2, and standard error of the regression manually (using numpy, not sklearn's `score`).
  - Use the fitted model to:
    - Construct a 95% confidence interval for the slope.
    - Perform a t-test for $H_0 : \beta_1 = 0$.
    - Plot regression line with data points and inspect residual vs fitted plot for nonlinearity.

8. Multiple regression and multicollinearity (H)

- **Dataset:** Same Boston Housing data: https://github.com/selva86/datasets/raw/master/BostonHousing.csv[2]
- **Task:**
  - Fit a multiple linear regression of `medv` on several predictors, e.g. `lstat`, `rm`, `ptratio`, `dis`.
  - Compute variance inflation factors (VIF) for each predictor to assess multicollinearity.
  - Compare standard errors and p-values with and without a highly correlated predictor (e.g. add `tax` or `rad`) and discuss how multicollinearity inflates uncertainty.

9. Nonparametric test: Mann–Whitney U (M/H)

- **Dataset:** Students Performance in Exams again: https://www.kaggle.com/datasets/spscientist/students-performance-in-exams[1]
- **Task:**
  - Compare `math score` distributions between `gender` groups using:
    - Two-sample t-test (assuming unequal variances).
    - Mann–Whitney U test (rank-based, nonparametric).
  - Check assumptions (normality, equal variance) with histograms and QQ-plots.
  - Discuss when the nonparametric result is more trustworthy than the t-test result.

10. ANOVA and post-hoc comparisons (H)

- **Dataset:** Iris dataset – standard CSV from UCI:
  https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data[3]
  (Column names: sepal_length, sepal_width, petal_length, petal_width, species.)
- **Task:**
  - Use one metric variable, e.g. `sepal_length`, and test whether its mean differs across the three `species` using one-way ANOVA.
  - If ANOVA is significant, do pairwise post-hoc comparisons (e.g. Tukey HSD via `statsmodels` or manual Bonferroni-corrected t-tests).
  - Visualize with boxplots per species and interpret which species differ and by how much, including effect sizes.

If you want, the next step can be to pick one question and get a skeleton Jupyter notebook / code template (imports, loading, plotting, and statistic formulas laid out) so you can fill in the reasoning yourself in Python.

⁂

1. https://www.kaggle.com/datasets?fileType=csv
2. https://github.com/MainakRepositor/Datasets
3. https://infoguides.gmu.edu/find-data/practice
4. https://www.dataquest.io/blog/free-datasets-for-projects/
5. https://www.kaggle.com/code/rtatman/fun-beginner-friendly-datasets
6. https://www.youtube.com/watch?v=krkS9u140tM
7. https://builtin.com/data-science/free-datasets
8. https://365datascience.com/tutorials/python-tutorials/free-public-datasets-python/
9. https://www.reddit.com/r/datascience/comments/y816ut/how_to_download_kaggle_datasets_using/
10. https://careerfoundry.com/en/blog/data-analytics/where-to-find-free-datasets/
11. https://www.datablist.com/learn/csv/download-sample-csv-files

12. https://www.reddit.com/r/dataanalysis/comments/18ozqzv/looking_for_raw_datasets_to_practice/
13. https://www.youtube.com/watch?v=qGu9pk63s8I
14. https://github.com/awesomedata/awesome-public-datasets
15. https://www.kaggle.com/datasets
16. https://mavenanalytics.io/data-playground
17. https://www.tableau.com/learn/articles/free-public-data-sets
18. https://careerbootcamps.tlcenter.wustl.edu/blog/data-science/free-public-data-sets-for-data-science-data-analytics-projects/
19. https://www.youtube.com/watch?v=jTTPmjtAY7o
20. https://datascientyst.com/search-download-kaggle-dataset-pandas-dataframe/